



AFRL-OSR-VA-TR-2013-0174

A Unifying High-Order Method for the Navier-Stokes Equations on Hybrid Unstructured Meshes

Z.J. Wang, T. Haga, H. Gao
Iowa State University

April 2013
Final Report

DISTRIBUTION A: Approved for public release.

AIR FORCE RESEARCH LABORATORY
AF OFFICE OF SCIENTIFIC RESEARCH (AFOSR)
ARLINGTON, VIRGINIA 22203
AIR FORCE MATERIEL COMMAND

REPORT DOCUMENTATION PAGE				<i>Form Approved OMB No. 0704-0188</i>	
The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to the Department of Defense, Executive Services and Communications Directorate (0704-0188). Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ORGANIZATION.					
1. REPORT DATE (DD-MM-YYYY) 19-02-2013		2. REPORT TYPE FINAL REPORT		3. DATES COVERED (From - To) March 1, 2009 - November 30, 2012	
4. TITLE AND SUBTITLE A Unifying High-Order Method for the Navier-Stokes Equations on Hybrid Unstructured Meshes				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER FA9550-09-1-0128	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S) Z.J. Wang T. Haga H. Gao				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Iowa State University 1138 Pearson Hall Ames, IA 50011				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) AFOSR 875 N Randolph St Arlington, VA 22203				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S) AFRL-OSR-VA-TR-2013-0174	
12. DISTRIBUTION/AVAILABILITY STATEMENT DISTRIBUTION A: APPROVED FOR PUBLIC RELEASE					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT This final report documents the major developments and findings during the grant period from March 2009 to November 2012. The main objective of this project was to develop a new discontinuous formulation named correction procedure via reconstruction (CPR) for hyperbolic conservation laws, and demonstrate its capability for the Euler and Navier-stokes equations on hybrid 3D unstructured prismatic and tetrahedral grids. We achieved the following accomplishments: • Extended the CPR formulation to 3D hybrid meshes, including tetrahedral, hexahedral, prismatic elements; • Extended the CPR formulation to the Navier-Stokes equations on hybrid elements, and demonstrate the method for benchmark 3D problems					
15. SUBJECT TERMS High-order methods, Computational fluid dynamics, unstructured meshes, Navier-Stokes equations					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT SAR	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON Z.J. Wang
a. REPORT U	b. ABSTRACT U	c. THIS PAGE U			19b. TELEPHONE NUMBER (Include area code) 785-864-2440

Reset

Final Report for Period March 2009 – November 2012
AFOSR Grant FA9550-09-1-0128

**A Unifying High-Order Method for the Navier-Stokes Equations
on Hybrid Unstructured Meshes**

Z.J. Wang^{1,2}, T. Haga¹, H. Gao¹

¹Department of Aerospace Engineering
Iowa State University, Ames, Iowa

²Department of Aerospace Engineering
University of Kansas, Lawrence, Kansas
zjw@ku.edu

February 19, 2013

Executive Summary

This final report documents the major developments and findings during the grant period from March 2009 to November 2012. The main objective of this project was to develop a new discontinuous formulation named correction procedure via reconstruction (CPR) for hyperbolic conservation laws, and demonstrate its capability for the Euler and Navier-stokes equations on hybrid 3D unstructured prismatic and tetrahedral grids. The CPR method can unify several popular high order methods including the discontinuous Galerkin, the spectral volume and spectral difference methods into an efficient differential form without explicit volume or surface integrals. By selecting the solution points to coincide with the flux points, solution reconstruction can be completely avoided.

We successfully fulfilled the main objective in the present study. More specifically, we achieved the following accomplishments:

- Extended the CPR formulation to 3D hybrid meshes, including tetrahedral, hexahedral, prismatic elements;
- Extended the CPR formulation to the Navier-Stokes equations on hybrid elements, and demonstrate the method for benchmark 3D problems
- Implemented the CPR method on clusters of CPUs and GPUs, and achieved up to two orders of magnitude speedup on the GPU than the CPU
- Extended the method for dynamic moving grids satisfying the so-called geometric conservation laws, and demonstrated the capability for bio-inspired flow problems
- Implemented solution-based hp-adaptations using a variety of adaptation criteria including residual, adjoint and entropy based adaptation criteria.

In the First International Workshop on High-Order CFD Methods, it was demonstrated through many benchmark cases that high-order methods are capable of achieving the same error with less CPU time than low order methods.

The grant supported 1 postdoc research associate, Dr. T. Haga, and several PhD students, H. Gao, Y. Zhou and L. Shi. The publications partly supported by the grant are listed below:

- Z.J. Wang, K.J. Fidkowski, R. Abgrall, F. Bassi, D. Caraeni, A. Cary, H. Deconinck, R. Hartmann, K. Hillewaert, H.T. Huynh, N. Kroll, G. May, P-O. Persson, B. van Leer, and M. Visbal. “High-Order CFD Methods: Current Status and Perspective,” International Journal for Numerical Methods in Fluids, 2012, Accepted.
- H. Gao and Z.J. Wang, “A Conservative Correction Procedure via Reconstruction Formulation with the Chain-Rule Divergence Evaluation”, J. Computational Physics 232, 7–13 (2013).
- Y. Li and Z.J. Wang, “An Optimized Correction Procedure via Reconstruction Formulation for Broadband Wave Computation”, Communications in Computational Physics, Vol. 13, No. 5, pp. 1265-1291 (2013).
- M. Yu, Z. J. Wang and H. Hu, “Airfoil Thickness Effects on the Thrust Generation of Plunging Airfoils”, J. of Aircraft, accepted.

- H. Gao, Z.J. Wang and H.T. Huynh, “Differential Formulation of Discontinuous Galerkin and Related Methods for the Navier-Stokes Equations”, *Communications in Computational Physics* 13, No. 4, 1013-1044 (2013).
- M. L. Yu, Z. J. Wang, H. Hu, “A High-Order Spectral Difference Method for Unstructured Dynamic Grids,” *Computers and Fluids* 48, 84–97 (2011).
- T. Haga, H. Gao and Z. J. Wang, “A High-Order Unifying Discontinuous Formulation for the Navier-Stokes Equations on 3D Mixed Grids,” *Math. Model. Nat. Phenom.*, Vol. 6 (3), 28-56 (2011).
- Y. Zhou and Z.J. Wang, “Absorbing boundary conditions for the Euler and Navier–Stokes equations with the spectral difference method”, *Journal of Computational Physics* 229, 8733–8749 (2010).
- H. Gao, Z.J. Wang, and Y. Liu, “A study of curved boundary representations for 2D high order Euler solvers,” *J. Scientific Computing* 44, 323–336, 2010.
- A.H. Mohammad, Z.J. Wang, C.L. Liang, “Large eddy simulation of flow over a cylinder using high-order spectral difference method,” *Advances in Applied Mathematics and Mechanics* 2 (4), 451-466, 2010.
- Z.J. Wang and Haiyang Gao, “A unifying lifting collocation penalty formulation including the discontinuous Galerkin, spectral volume/difference methods for conservation laws on mixed grids,” *Journal of Computational Physics* 228 (2009) 8161–8186.
- R. Harris and Z.J. Wang, “High-order adaptive quadrature-free spectral volume method on unstructured grids,” *Computers & Fluids* 38 (2009) 2006–2025.
- M. Yang and Z.J. Wang, "A Parameter-Free Generalized Moment Limiter for High-Order Methods on Unstructured Grids," *Advances in Applied Mathematics and Mechanics* 1 (2009), 451-480.
- C. Liang, A. Jameson, and Z.J. Wang, “Spectral difference method for compressible flow on unstructured grids with mixed elements,” *Journal of Computational Physics*, 228 (2009) 2847–2858.
- Y. Sun, Z.J. Wang and Y. Liu, “Efficient Implicit Non-linear LU-SGS Approach for Compressible Flow Computation Using High-Order Spectral Difference Method”, *Communications in Computational Physics*, Vol. 5, No. 2-4, pp. 760-778 (2009).
- C. Liang, R. Kannan and Z.J. Wang, “A p-multigrid Spectral Difference method with explicit and implicit smoothers on unstructured triangular grids,” *Computers & Fluids*, 38, 254–265 (2009).

1. Introduction

Advantages of high-order methods are well recognized in the computational fluid dynamics (CFD) community especially for aeroacoustic noise predictions, vortex dominated flows, large eddy simulation and direct numerical simulation (DNS) of turbulent flows. Since the truncation error of a high-order method decreases more rapidly than that of a lower order method, the more stringent the accuracy requirement is, the more efficient a high-order method becomes in computational cost. For the practical use in industries, lower order (1st or 2nd) unstructured grid methods are usually employed for the reason of superior geometrical flexibility and robustness. However, these methods are likely too dissipative to capture small vortex structures in turbulent flows and are often not capable of obtaining grid converged solutions. Increased prediction accuracy is often required for many aerodynamic problems with both complex physics and geometry, such as helicopter blade vortex interactions, flow over high lift devices, and aeroacoustic noise generated by the landing gear.

In the past decades, there has been significant progress in developing high-order methods capable of solving the Navier-Stokes (NS) equations on unstructured grids. For compressible flow computations in aerospace applications, the discontinuous Galerkin (DG) method [31,4,5,7,1,2,29,42] has attracted intensive interest. One particular feature of the DG method is the discontinuous solution space of high-order approximations for each element, which allows the scheme to be very flexible in dealing with complex configuration and in accommodating solution based adaptations. Other methods assuming element-wise discontinuous solution are staggered-grid (SG) multi-domain spectral method [21], spectral volume (SV) [43,46-48,25,39,14,12] and spectral difference (SD) [23,24,28,40] methods. Another notable feature that is common among these methods is the use of one of the Riemann solvers [33,32,30,19,22] to compute unique fluxes at element interfaces to incorporate “upwinding” characteristics of wave propagation, similar to the Godunov type finite volume method [11,41]. The main difference among these methods lies in how the governing equations are discretized and the degrees-of-freedom (DOFs) are chosen. The DG method is based on the weighted residual form. Various types of DG schemes are derived with different choice of DOFs. Depending on how the DOFs are defined, DG schemes can be further divided into modal and nodal approaches. The SV method is discretized in the integral form similar to the finite volume method and the DOFs are always the sub-cell or control volume (CV) averages. The SG/SD method is based on the differential form without any integration and the DOFs are chosen as the nodal values within each element. More comprehensive reviews of these methods are given in [44].

Recently, a novel formulation named CPR (correction procedure via reconstruction) was developed by Huynh [17,18] for 1D conservation laws, and extended to simplex and hybrid meshes by Wang and Gao [45]. The CPR method is based on a nodal differential form, with an element-wise discontinuous polynomial solution space. The solution polynomial is interpolated from the solutions at a set of solution points. This formulation has some remarkable properties. The framework is easy to understand, efficient to implement and recovers several known methods such as the DG, SG or the SV/SD methods. Furthermore, by choosing the solution points to coincide with the flux points, the reconstruction of solution polynomials to calculate the residual can be completely avoided. The DG scheme derived through the CPR framework is probably the simplest and most efficient amongst all DG formulations since explicit integrations are avoided. In a recent study [9], the CPR method has been extended to the Navier-Stokes

equations on 2D mixed meshes. These successful developments laid a solid foundation for its efficient implementation and demonstration on arbitrary grids in 3D.

Hybrid elements such as hex, prism, pyramid and tetrahedron will provide great geometrical flexibility for practical problems in 3D. In particular, for high Reynolds number flows in aerodynamic applications, prismatic cells have the advantages in accuracy and computational cost to resolve boundary layers near the wall. There have been several attempts to develop the DG method on arbitrary grid elements. In [34] different types of elements such as hex, prism and pyramid are projected onto a reference cube using collapsed Cartesian coordinates and hierarchical basis functions over the cube are used. Luo et.al. [26] presented a different approach based on the Taylor series expansion at the center of arbitrary element. Gassner et.al. [10] used polymorphic nodal element in the modal based formulation to reduce the cost of numerical integrations. However one obvious shortcoming of these formulations is the high computational cost of the surface and volume integrations coming from the weighted residual formulation. Another difficulty is the treatment of curvilinear boundary elements. The simple formulation of the CPR method is expected to alleviate the computational costs and facilitate the treatment of curved wall surfaces in a straightforward fashion.

In the present study, we develop the CPR for solving the Euler and Navier-Stokes equations on 3D mixed meshes. For the current implementation, tetrahedral and prismatic elements are considered with the intention to resolve viscous boundary layer flows efficiently. The remainder of this article is organized as follows. The basic formulation of the CPR method is described in the next section. In section 3, The discretization of the compressible Navier-Stokes equations is derived in the CPR framework. Subsequently, we discuss how to implement the CPR method efficiently in each particular element with curvilinear geometry in section 4. Section 5 presents the computational results for several benchmark problems, including accuracy studies on mixed unstructured grids. Conclusions for the present study and possible future works are summarize in section 6.

2. Review of the Correction Procedure via Reconstruction Formulation

We first review the CPR formulation for a hyperbolic conservation law, which can be written as

$$\frac{\partial Q}{\partial t} + \nabla \cdot \vec{F}(Q) = 0, \quad (2.1)$$

with suitable initial and boundary conditions. Q is the vector of conserved variables, and \vec{F} is the flux vector. Assume that the computational domain is discretized into N non-overlapping elements $\{V_i\}$. The weighted residual form of (2.1) on element V_i can be derived by multiplying (2.1) by an arbitrary weighting or test function W and integrating over V_i ,

$$\int_{V_i} \frac{\partial Q}{\partial t} W dV + \int_{\partial V_i} W \vec{F}(Q) \cdot \vec{n} dS - \int_{V_i} \nabla W \cdot \vec{F}(Q) dV = 0. \quad (2.2)$$

Let Q_i^h be an approximate solution to Q at element V_i . We assume that the solution belongs to the space of polynomials of degree k or less, i.e., $Q_i^h \in P^k$, within each element without continuity

requirement across element interfaces. Then, we require that the numerical solution Q_i^h must satisfy (2.2), i.e.,

$$\int_{V_i} \frac{\partial Q_i^h}{\partial t} W dV + \int_{\partial V_i} W \vec{F}(Q_i^h) \cdot \vec{n} dS - \int_{V_i} \nabla W \cdot \vec{F}(Q_i^h) dW = 0. \quad (2.3)$$

Because the approximated solution is in general discontinuous across element interfaces, the fluxes at the interfaces are not well defined. To evaluate a unique flux and also to provide element coupling, a common Riemann flux is used to replace the normal flux, i.e.,

$$F^n(Q_i^h) \equiv \vec{F}(Q_i^h) \cdot \vec{n} \approx \tilde{F}^n(Q_i^h, Q_{i+}^h, \vec{n}), \quad (2.4)$$

where Q_{i+}^h is the solution from outside of the current element V_i . Thus, Eq. (2.3) becomes

$$\int_{V_i} \frac{\partial Q_i^h}{\partial t} W dV + \int_{\partial V_i} W \tilde{F}^n(Q_i^h, Q_{i+}^h, \vec{n}) dS - \int_{V_i} \nabla W \cdot \vec{F}(Q_i^h) dW = 0. \quad (2.5)$$

If the space of W is chosen to be the same as the solution space, Eq. (2.5) is equivalent to the DG formulation. For the sake of a much simpler formulation, we wish to eliminate the test function. Applying integration by parts to the last term of (2.5), we obtain

$$\int_{V_i} \frac{\partial Q_i^h}{\partial t} W dV + \int_{V_i} W \nabla \cdot \vec{F}(Q_i^h) dV + \int_{\partial V_i} W [\tilde{F}^n(Q_i^h, Q_{i+}^h, \vec{n}) - F^n(Q_i^h)] dS = 0. \quad (2.6)$$

Note that the surface integral coming from the last term of (2.5) is evaluated by only using the internal solution. The last term of (2.6) can be viewed as a penalty term, i.e., penalizing the normal flux differences $[\tilde{F}] \equiv \tilde{F}^n(Q_i^h, Q_{i+}^h, \vec{n}) - F^n(Q_i^h)$. Let us introduce a “correction field” $\delta_i \in P^k$, which is determined from the following relation defining the so-called “lifting operator” for $[\tilde{F}]$.

$$\int_{V_i} W \delta_i dV = \int_{\partial V_i} W [\tilde{F}] dS. \quad (2.7)$$

Substituting (2.7) into (2.6), we obtain

$$\int_{V_i} \left[\frac{\partial Q_i^h}{\partial t} + \nabla \cdot \vec{F}(Q_i^h) + \delta_i \right] W dV = 0. \quad (3.8)$$

In the present study, in order to simplify the derivation we also approximate the flux divergence by polynomials of degree k or less, i.e. $\nabla \cdot \vec{F}(Q_i^h) \in P^k$. If W is selected such that a unique solution exists, (3.8) is equivalent to

$$\frac{\partial Q_i^h}{\partial t} + \nabla \cdot \vec{F}(Q_i^h) + \delta_i = 0, \quad (2.9)$$

i.e., (2.9) is satisfied everywhere in element V_i . With the definition of a correction field, we have successfully reduced the weighted residual formulation to an equivalent simple differential form, which does not involve any explicit surface or volume integrals.

To find the approximate solution Q_i^h , let the DOFs be the solution values at a set of points $\{r_{i,j}\}$, named solution points (SPs). Then equation (2.9) must hold at the SPs, i.e.,

$$\frac{\partial Q_{i,j}^h}{\partial t} + \nabla \cdot \vec{F}(Q_{i,j}^h) + \delta_{i,j} = 0. \quad (2.10)$$

Obviously the correction field δ_i can be expressed in terms of $\delta_{i,j}$ using a Lagrange interpolation on the SPs, i.e.,

$$\delta_i = \sum_j L_j^{\text{SP}}(\vec{r}_{i,j}) \delta_{i,j}, \quad (2.11)$$

where $L^{\text{SP}} \in P^k$ are the Lagrange polynomials based on the SPs. In the case of a non-linear flux vector, $\vec{F}(Q_i^h)$ are not polynomials in general. In the present study, we approximate $\vec{F}(Q_i^h)$ by polynomials of degree k to evaluate RHS of eq. (2.7). Thereby, we assume that the flux difference $[\tilde{F}]$ is a polynomial on the faces of the element, and can be determined based on values of $[\tilde{F}]_{f,l}$ at a set of flux points (FPs) $\{r_{f,l}\}$ using a Lagrange interpolation, i.e.,

$$[\tilde{F}]_f = \sum_l L_l^{\text{FP}}(\vec{r}_{f,l}) [\tilde{F}]_{f,l}, \quad (2.12)$$

where $L^{\text{FP}} \in P^k$ are the Lagrange polynomials based on the FPs. Then, if the locations of the solution and flux points are specified and the weighting function W is chosen so as to have the same dimension as the correction field δ_i , $\delta_{i,j}$ can be uniquely defined by solving the linear system derived from eq. (2.7). For simplex elements with straight faces, it can be expressed in the following formula

$$\delta_{i,j} = \frac{1}{|V_i|} \sum_{f \in \partial V_i} \sum_l \alpha_{j,f,l} [\tilde{F}]_{f,l} S_f, \quad (2.13)$$

where $\alpha_{j,f,l}$ are constant coefficients independent of the solution. Substituting (2.13) into (2.10) we obtain the following equation

$$\frac{\partial Q_{i,j}^h}{\partial t} + \nabla \cdot \vec{F}(Q_{i,j}^h) + \frac{1}{|V_i|} \sum_{f \in \partial V_i} \sum_l \alpha_{j,f,l} [\tilde{F}]_{f,l} S_f = 0. \quad (2.14)$$

One can clearly see that this is a collocation-like formulation with penalty-like term that comes from the element-wise correction polynomial to provide the coupling between elements. It can be shown that the location of SPs does not affect the numerical scheme for linear conservation laws [40,17]. For efficiency, the solution points are always chosen to coincide with the flux points. Therefore, any data interpolation is no longer needed for flux calculation, which dramatically reduces the computational cost. Any convergent nodal sets with enough points at the element interface are good candidates, e.g., those found in [3,15,49].

Finally we want to make a remark on the relationship between the CPR formulation and other methods including DG, SV and SD methods. Starting from the weighted residual form of the governing equations, different formulations can be derived depending on the weighting function. For example, a nodal DG formulation is obtained by choosing weighting functions to be Lagrange polynomials, and a SV formulation is obtained by defining weighting functions as piecewise constant at the sub-cells. As a result, the only difference between those schemes appears in the correction coefficients. In the original work [45], it was shown that the resulting CPR scheme is basically conservative by using the correction coefficients for the DG, SV and SD scheme. In this study, we choose the weighting function to be one of the Lagrange polynomials based on the SPs, i.e., eq. (2.9) is identical to the DG formulation.

3. Discretization of the Navier-Stokes Equations

3.1. Governing Equations

The 3D compressible Navier-Stokes equations can be written as a system of partial differential equations in conservation form:

$$\frac{\partial Q}{\partial t} + \nabla \cdot (\bar{F}_c(Q) - \bar{F}_v(Q, \nabla Q)) = 0, \quad (3.1)$$

where Q , $\bar{F}_c = [F_c^x, F_c^y, F_c^z]$ and $\bar{F}_v = [F_v^x, F_v^y, F_v^z]$ denote the conservative state vector, the inviscid and the viscous flux vectors, respectively, and are given by

$$Q = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ e \end{pmatrix}, \quad F_c^x = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ (e+p)u \end{pmatrix}, \quad F_c^y = \begin{pmatrix} \rho v \\ \rho v^2 + p \\ \rho uv \\ \rho vw \\ (e+p)v \end{pmatrix}, \quad F_c^z = \begin{pmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ (e+p)w \end{pmatrix} \quad (3.2)$$

$$F_v^x = \begin{pmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ \tau_{xz} \\ u\tau_{xx} + v\tau_{xy} + w\tau_{xz} - q_x \end{pmatrix}, \quad F_v^y = \begin{pmatrix} 0 \\ \tau_{yx} \\ \tau_{yy} \\ \tau_{yz} \\ u\tau_{yx} + v\tau_{yy} + w\tau_{yz} - q_y \end{pmatrix}, \quad F_v^z = \begin{pmatrix} 0 \\ \tau_{zx} \\ \tau_{zy} \\ \tau_{zz} \\ u\tau_{zx} + v\tau_{zy} + w\tau_{zz} - q_z \end{pmatrix} \quad (3.3)$$

where ρ is the density, $\vec{v} = [u, v, w]$ are the velocity vector, p is the pressure, e is the total energy per unit volume. The viscous stress tensor can be represented as

$$\tau = \mu \left(\nabla \vec{v} + (\nabla \vec{v})^T - \frac{2}{3} (\nabla \cdot \vec{v}) I \right) \quad (3.4)$$

where μ is the molecular viscosity coefficient, I is the unit tensor. The heat flux is given as

$$\vec{q} = -c_p \frac{\mu}{Pr} \nabla T \quad (3.5)$$

Here, c_p is the specific heat capacity at constant pressure and T is the temperature. The Prandtl number Pr is assumed to be a constant of 0.72 in this study. For a perfect gas, the pressure is related to the total energy e by

$$e = \frac{p}{\gamma-1} + \frac{1}{2} \rho (u^2 + v^2 + w^2) \quad (3.6)$$

The specific heat ratio γ is set to be a constant, 1.4 for air. The computations for solving the Euler equations are performed by omitting the viscous flux.

3.2. CPR Formulation of the Navier-Stokes Equations

In order to discretize the Navier-Stokes equations, we follow a mixed formulation that is commonly used for the DG method[2, 6]. By introducing a new variable $R = \nabla Q$, Eq. (3.1) is rewritten in a first order system as

$$\frac{\partial Q}{\partial t} + \nabla \cdot (F_c(Q) - F_v(Q, R)) = 0, \quad (3.7)$$

$$R = \nabla Q. \quad (3.8)$$

According to the CPR formulation by assuming $Q_i^h, R_i^h \in P^k$ on discretized elements $\{V_i\}$, we obtain

$$\frac{\partial Q_{i,j}^h}{\partial t} + \nabla \cdot (\tilde{F}_c(Q_{i,j}^h) - \tilde{F}_v(Q_{i,j}^h, R_{i,j}^h)) + \frac{1}{|V_i|} \sum_{f \in \partial V_i} \sum_l \alpha_{j,f,l} ([\tilde{F}_c]_{f,l} - [\tilde{F}_v]_{f,l}) S_f = 0. \quad (3.9)$$

$$R_{i,j}^h = (\nabla Q_i^h)_j + \frac{1}{|V_i|} \sum_{f \in \partial V_i} \sum_l \alpha_{j,f,l} [\tilde{Q}]_{f,l} \tilde{n}_f S_f, \quad (3.10)$$

where $[\tilde{F}_c] \equiv \tilde{F}_c^n(Q_i^h, Q_{i+}^h, \tilde{n}) - F_c^n(Q_i^h)$, $[\tilde{F}_v] \equiv \tilde{F}_v^n(Q_i^h, Q_{i+}^h, \nabla Q_i^h, \nabla Q_{i+}^h, \tilde{n}) - F_v^n(Q_i^h, R_i^h)$ and $[\tilde{Q}] \equiv \tilde{Q}(Q_i^h, Q_{i+}^h) - Q_i^h$.

3.2.1. Inviscid Flux Calculation

Here we consider the inviscid flux. We need to discretize the internal flux divergence and the common flux at the interface in (3.9). Instead of approximating the inviscid flux by the Lagrange interpolation on the SPs, the flux divergence is calculated “exactly” at the solution points with the chain rule (CR) approach

$$\nabla \cdot \tilde{F}_c(Q_{i,j}^h) = \frac{\partial \tilde{F}_c(Q_{i,j}^h)}{\partial Q} \cdot \nabla Q_{i,j}^h, \quad (3.11)$$

where $\frac{\partial \bar{F}_c}{\partial Q}$ is the inviscid flux Jacobian matrix. Note that $\nabla \bullet \bar{F}_c(Q_i^h)$ is generally not a degree k polynomial, but it can be approximated by the Lagrange polynomial of degree k using the flux divergence at the solution points, i.e.,

$$\nabla \bullet \bar{F}_c(Q_i^h) \approx \nabla \bullet \bar{F}_c^{\text{CR}}(Q_i^h) = \sum_j L_j^{SP}(\bar{r}_{i,j}) \nabla \bullet \bar{F}_c(Q_{i,j}^h), \quad (3.12)$$

This implies that the flux vector $\bar{F}_c^{\text{CR}}(Q_i^h)$ belongs to P^{k+1} which is one degree higher than the approximation of $\bar{F}(Q_i^h) \in P^k$ used in the correction term. The chain rule approach is known to be more accurate, though the resulting scheme is not strictly conservative due to the inconsistency between the approximated flux vectors in the flux divergence term and the correction term. However, it is also known that the mass conservation error is extremely small [45]. If conservation is absolutely required, one can use the Lagrange polynomial approach [17].

The common inviscid flux in $[\tilde{F}_c]$ can be obtained with any Riemann solver. In this paper, the Roe flux [32] is used for all the cases.

3.2.2. Viscous Flux Calculation

In the present study, we employ the BR2 scheme [2] to discretize the viscous flux. In (3.10), the common solution $\tilde{Q}_{f,l}$ is simply the average of the solutions at both sides of f . The viscous fluxes at the solution points are evaluated by $F^v(Q_{i,j}^h, R_{i,j}^h)$. Then the viscous flux divergence is obtained through the Lagrange interpolation rather than the CR approach. In the correction term, the common viscous flux also needs to be determined. Besides the common solution, we also need to define a common gradient on face f . The common gradient is evaluated as

$$\nabla Q_{f,l}^{\text{com}} = \frac{1}{2} (\nabla Q_{f,l}^- + r_{f,l}^- + \nabla Q_{f,l}^+ + r_{f,l}^+) \quad (3.13)$$

where $\nabla Q_{f,l}^-$ and $\nabla Q_{f,l}^+$ are the gradients of the solution from the left and right cells, while $r_{f,l}^-$ and $r_{f,l}^+$ are the local lifting correction to the gradients only due to the common solution on face f

$$r_{f,l}^\pm = \frac{1}{|V^\pm|} \sum_m \alpha_{l,f,m} [\tilde{Q}]_{f,m}^\pm (\mp \bar{n}_f) \mathcal{F}_f, \quad (3.14)$$

where m is the index for the flux points on f and \bar{n}_f is the unit normal vector directing from left to right. Note that there is no summation over all faces of the element in eq. (3.14) in order to assure local property of the BR2 scheme.

4. Discretization on Mixed Grids with Curved Boundary

It can be observed that (2.9) is valid for arbitrary types of elements besides triangles and tetrahedrons. The current development for 3D hybrid meshes accommodates two kinds of element shapes, i.e., tetrahedron and triangular prism. Other types of element such as hexahedron and pyramid will be developed in the near future. The use of prismatic cells in addition to tetrahedral cells has the advantages in both accuracy and computational costs to resolve boundary layers near solid walls. In order to achieve an efficient implementation, all elements

are transformed from the physical domain (x, y, z) into a corresponding standard element in the computational domain (ξ, η, ζ) as shown in Fig. 1. Here we consider the transformations for the elements with curved sides (faces and edges). The discretization for the curved elements is conducted in the same way as the straight sided elements by applying the CPR formulation in the standard elements. In the present study, a quadratic triangular face is employed to represent curved wall boundaries. For the sake of computational efficiency, the quadratic representation is adopted for only one of the faces of tetrahedron which will be attached to the wall in inviscid flows, and for only two triangular faces of prism which will be used in the thin layers of prism cells to assure the quality of the element shape especially in high Reynolds number flows.

Based on a set of locations of nodes defining the shape of element, a set of shape functions can be obtained [50]. Once the shape functions $M_i(\xi, \eta, \zeta)$ are given, the transformation can be written as

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \sum_{i=1}^K M_i(\xi, \eta, \zeta) \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix}, \quad (4.1)$$

where K is the number of points used to define the physical element, (x_i, y_i, z_i) are the Cartesian coordinates of those points. For the transformation given in (4.1), the Jacobian matrix J takes the following form

$$J = \frac{\partial(x, y, z)}{\partial(\xi, \eta, \zeta)} = \begin{bmatrix} x_{\xi} & x_{\eta} & x_{\zeta} \\ y_{\xi} & y_{\eta} & y_{\zeta} \\ z_{\xi} & z_{\eta} & z_{\zeta} \end{bmatrix}. \quad (4.2)$$

For a non-singular transformation, its inverse transformation must also exist, and the Jacobian matrices are related to each other according to

$$\frac{\partial(\xi, \eta, \zeta)}{\partial(x, y, z)} = \begin{bmatrix} \xi_x & \xi_y & \xi_z \\ \eta_x & \eta_y & \eta_z \\ \zeta_x & \zeta_y & \zeta_z \end{bmatrix} = J^{-1}. \quad (4.3)$$

The governing equations in the physical domain are then transformed into the computational domain (standard element), and the transformed equations take the following form

$$\frac{\partial \tilde{Q}}{\partial t} + \frac{\partial F^{\xi}}{\partial \xi} + \frac{\partial F^{\eta}}{\partial \eta} + \frac{\partial F^{\zeta}}{\partial \zeta} = 0, \quad (4.4)$$

where

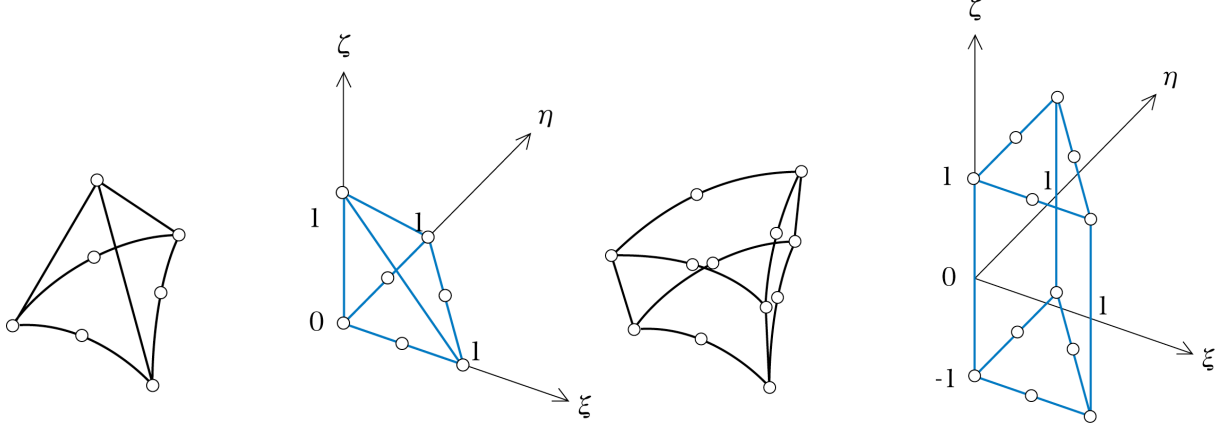


Figure 1. Transformations of a curve boundary tetrahedral and prismatic cell to the standard

$$\begin{aligned}
 \tilde{Q} &= |J| \cdot Q, \\
 F^\xi &= |J| \cdot (\xi_x F^x + \xi_y F^y + \xi_z F^z) \\
 F^\eta &= |J| \cdot (\eta_x F^x + \eta_y F^y + \eta_z F^z) \\
 F^\zeta &= |J| \cdot (\zeta_x F^x + \zeta_y F^y + \zeta_z F^z)
 \end{aligned} \tag{4.5}$$

Let $\tilde{S}_\xi = |J|(\xi_x, \xi_y, \xi_z)$, $\tilde{S}_\eta = |J|(\eta_x, \eta_y, \eta_z)$ and $\tilde{S}_\zeta = |J|(\zeta_x, \zeta_y, \zeta_z)$. Then we have $F^\xi = \vec{F} \cdot \tilde{S}_\xi$, $F^\eta = \vec{F} \cdot \tilde{S}_\eta$ and $F^\zeta = \vec{F} \cdot \tilde{S}_\zeta$. In our implementation, $|J|$, \tilde{S}_ξ , \tilde{S}_η and \tilde{S}_ζ are stored at the solution points. Note that here we consider the Euler equations as the governing equations for brevity's sake. Extending the following discretization to the Navier-Stokes equations is straightforward.

Discretization on a Standard Tetrahedron

On a standard tetrahedron, the CPR formulation in eq. (2.14) can be rewritten as

$$\frac{\partial \tilde{Q}_{i,j}^h}{\partial t} + \nabla^{(\xi)} \cdot \vec{F}^{(\xi)}(\tilde{Q}_{i,j}^h) + \frac{1}{|V^{(\xi)}|} \sum_{f \in \partial V} \sum_l \alpha_{j,f,l} [\tilde{F}^{(\xi)}]_{f,l} S_f^{(\xi)} = 0, \tag{4.6}$$

where superscript (ξ) means the variables or operations evaluated on the computational domain. For example, $[\tilde{F}^{(\xi)}]$ are the normal jumps of the transformed fluxes across the faces of the standard element. The transformed normal flux can be expressed in terms of the flux in the physical space as

$$\begin{aligned}
 \vec{F}_n^{(\xi)}|_{f,l} &= \vec{F}^{(\xi)}|_{f,l} \cdot \vec{n}^{(\xi)}|_f \\
 &= \vec{F}|_{f,l} \cdot \tilde{S}_\xi|_{f,l} n^\xi|_f + \vec{F}|_{f,l} \cdot \tilde{S}_\eta|_{f,l} n^\eta|_f + \vec{F}|_{f,l} \cdot \tilde{S}_\zeta|_{f,l} n^\zeta|_f \\
 &= \vec{F}|_{f,l} \cdot \tilde{S}_n|_{f,l} = \vec{F}_n|_{f,l} \cdot \tilde{S}_n|_{f,l},
 \end{aligned} \tag{4.7}$$

where $\vec{n}^{(\xi)} = [n^\xi, n^\eta, n^\zeta]$ is a unit normal vector on a straight face of the standard element, and \vec{S}_n is a normal vector on a face in the physical space defined as $\vec{S}_n = \vec{S}_\xi n^\xi + \vec{S}_\eta n^\eta + \vec{S}_\zeta n^\zeta$.

Note that solving Eq. (4.4), $\tilde{Q} = |J| \cdot Q$ are the solution unknowns, and are assumed to be degree k polynomials in the computational domain instead of Q . As a result, the derivatives of Q should be calculated in the following way,

$$\frac{\partial Q}{\partial \xi} = \frac{1}{|J|} \left[\frac{\partial(|J|Q)}{\partial \xi} - \frac{\partial|J|}{\partial \xi} Q \right], \quad \frac{\partial Q}{\partial \eta} = \frac{1}{|J|} \left[\frac{\partial(|J|Q)}{\partial \eta} - \frac{\partial|J|}{\partial \eta} Q \right], \quad \frac{\partial Q}{\partial \zeta} = \frac{1}{|J|} \left[\frac{\partial(|J|Q)}{\partial \zeta} - \frac{\partial|J|}{\partial \zeta} Q \right]. \quad (4.8)$$

In 3D, to construct a complete polynomial of degree k , at least $k(k+1)(k+2)/(1 \times 2 \times 3)$ SPs need to be chosen. In order to achieve the most efficient implementation, SPs on edges are chosen to be the Legendre-Gauss Lobatto (LGL) points. For 4th- or higher order schemes, nodes inside the boundary triangle are chosen from [15]. For 5th- or higher order schemes, nodes inside the tetrahedron are chosen from [49]. The nodal set of the 4th-order CPR scheme is shown in Fig. 2. Note that the flux difference at a flux point corrects all solution points as shown in (4.6).

Discretization on Standard Prism

For a standard triangular prism, the solution polynomial can be expressed as a tensor product of a 1D and 2D Lagrange polynomial, i.e.,

$$\tilde{Q}_i^h(\xi, \eta, \zeta) = \sum_{k=1} \sum_{j=1} \tilde{Q}_{i,j,k}^h L_j(\xi, \eta) L_k(\zeta), \quad (4.9)$$

where $\tilde{Q}_{i,j,k}^h$ are the state variables at the solution point (j,k) , with j the index in ξ - η plane and k the index in ζ direction, $L_j(\xi, \eta)$ is a 2D Lagrange polynomial in a triangle and $L_k(\zeta)$ is a 1D Lagrange polynomial in a segment. Figure 3 shows the locations of the solution points for $k=3$. The nodal sets on the edge and the triangle are chosen in the same manner with the tetrahedral element.

For the extension of the CPR method to 2D quadratic elements [17], the solution procedure reduces to a series of one-dimensional operations, i.e., the solution polynomial is represented as a tensor product of 1D Lagrange polynomials and the correction due to flux difference is performed in a one-dimensional manner. The CPR formulation for a standard prism can be derived in an analogous fashion as

$$\begin{aligned} & \frac{\partial \tilde{Q}_{i,j,k}^h}{\partial t} + \nabla^{(\xi)} \bullet \vec{F}^{(\xi)}(\tilde{Q}_{i,j,k}^h) + \frac{1}{|V_{Tri}^{(\xi)}|} \sum_{f \in \partial V_{Tri}} \sum_l \alpha_{j,f,l} [\tilde{F}^{(\xi)}(\xi_{f,l}, \eta_{f,l}, \zeta_k)] S_f^{(\xi)} \\ & + [\tilde{F}^\zeta(\xi_j, \eta_j, -1) - F^\zeta(\xi_j, \eta_j, -1)] g'_L(\zeta_k) + [\tilde{F}^\zeta(\xi_j, \eta_j, 1) - F^\zeta(\xi_j, \eta_j, 1)] g'_R(\zeta_k) = 0. \end{aligned} \quad (4.10)$$

The correction process is done in a decoupled manner. The third term is the correction of the flux components in ξ and η direction, which is computed on a plane with fixed $\zeta = \zeta_k$. This is nothing but the correction used in the 2D CPR method for a triangular element. In eq. (4.10), V_{Tri} is the area of triangle, S_f the length of the edge f and l the index for flux points on f . Note that, $[\tilde{F}(\xi_{f,l}, \eta_{f,l}, \zeta_k)]$ corrects only the solution points on the triangle with fixed k instead of all solution points in the element as shown in Fig. 3 (a). The last two terms denote the correction in the ζ

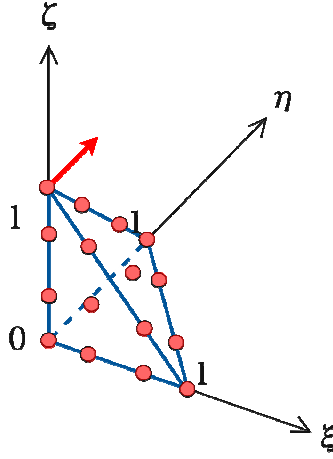


Figure 2. Solution points in the standard tetrahedral cell for degree $k=3$ polynomial (only points on the visible faces are shown).

direction, which is evaluated by the 1D CPR method [17]. g_L and g_R are the correction functions for the left and right end points of a segment. The flux difference at an end point corrects only the solution points on the segment with fixed j as shown in Fig. 3 (b). For prism cells, the number of solution points corrected by a flux point is smaller than the one for tetrahedral cells due to the decoupled correction procedure. Hence, the method for prisms is more efficient per DOF than for tetrahedrons. This decoupled procedure also facilitates the implementation employing different degrees of polynomials in ξ - η and ζ directions to adapt to flow features. An attempt employing higher order polynomials in the wall normal direction to resolve the boundary layer with coarser prism cells is shown in a later section.

In order to simplify the implementation for mixed grids, we assume the polynomial degree k to be the same for both the tetrahedral and prismatic elements. Furthermore, the flux points along the element interfaces are required to match each other. In the present implementation, the flux points are selected to be the LGL points at each edge for all tetrahedral and prismatic elements.

5. Numerical Results

5.1. Test cases for the Euler Equations

5.1.1. Accuracy Study with Vortex Evolution Problem

To assess the order of accuracy of the developed method, the propagation of an isentropic vortex in inviscid flow is computed with successive grid refinement. This is an idealized problem for the Euler equations in 2D used by Shu [35]. Here we consider simple extension of this problem to the 3D domain $[0, 10] \times [0, 10] \times [0, 10]$. The mean flow is $\{\rho, u, v, w, p\} = \{1, 1, 1, 0, 1\}$. An isotropic vortex is then added to the mean flow, i.e., with perturbations in u, v , and temperature $T = p/\rho$, and no perturbation in entropy $S = p/\rho^\gamma$:

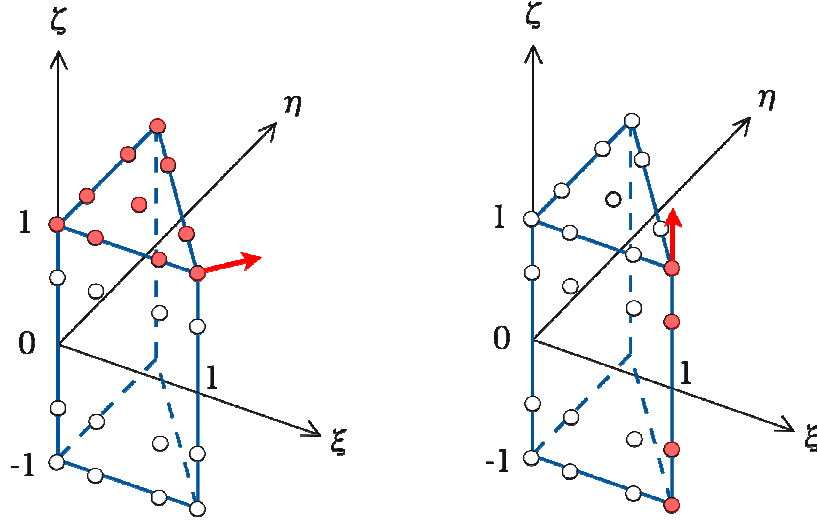


Figure 3. Solution points in the standard prism cell for degree $k=3$ polynomial (only points on the visible faces are shown). (a) shows the correction in the ξ and η directions. (b) shows the

$$\begin{aligned} \delta u &= -\bar{y} \frac{\varepsilon}{2\pi} e^{0.5(1-\bar{r}^2)}, \quad \delta v = \bar{x} \frac{\varepsilon}{2\pi} e^{0.5(1-\bar{r}^2)}, \quad \delta w = 0 \\ \delta T &= -\frac{(\gamma-1)\varepsilon^2}{8\gamma\pi^2} e^{1-\bar{r}^2}, \quad \delta S = 0, \end{aligned} \quad (5.1)$$

where $\bar{r}^2 = \bar{x}^2 + \bar{y}^2$, $\bar{x} = x - 5$, $\bar{y} = y - 5$, and the vortex strength $\varepsilon = 5$. If the computational domain is infinitely big, the exact solution of the Euler equations with the above initial condition is just the passive convection of the isentropic vortex with the mean velocity $(1, 1, 0)$. In the numerical simulation, we impose the exact solution on the boundaries.

The computations are carried out until $t=2$ on two different types of grids, tetrahedral meshes and prismatic meshes. In generating computational grids, first an equidistant Cartesian grid of $N \times N \times N$ cells is assumed for the cubic domain and each cell is further divided into six tetrahedrons or two prisms. Three different grids are employed with $N=10, 20$ and 40 for each type of cell. For the time integration, a 3rd-order Runge-Kutta explicit scheme is used. The L_1

and L_∞ norms of density error at the solution points are presented for tetrahedral grids and prismatic grids in Table 1 and 2, respectively. The CPR-DG method performs very well on both types of grid, achieving the nearly optimal order of accuracy up to 6th-order in tetrahedral meshes and 4th-order in prismatic meshes.

5.1.2. Subsonic Inviscid Flow over a Sphere

In order to verify the developed Euler solver on a mixed mesh with curved wall boundary, a typical steady test case of a subsonic flow around a sphere is considered. The freestream Mach number is $M=0.3$. Two computational grids are employed. One is a purely prismatic grid and the other is a mixed grid shown in Figures 4 (a) and 5 (a). The mixed grid is composed of five layers of prismatic cells around the quarter sphere and isotropic tetrahedral cells for the remaining region. To preserve the geometry of the sphere well with a relatively coarse mesh, the curved wall boundaries are represented by quadratic polynomials.

The computed density contours obtained with the 2nd- to 4th-order schemes are shown at Figure 4 (b)-(d) and Figure 5 (b)-(d). In both grids, the trends of improvement in the solution by increasing the order of discretization are similar. The computed density contours using the 4th order scheme appear to be perfectly symmetric without visible numerical dissipation and also quite smooth across the interface between prismatic and tetrahedral cells. In this case, a block LU-SGS implicit scheme [36,13] was used to obtain steady solutions efficiently, and all the cases converged to machine zero.

5.2. Test cases for the Navier-Stokes Equations

5.2.1. Accuracy Study with Couette Flow Problem

A laminar flow between two parallel walls is considered here to verify the discretization of viscous effects. The distance between the walls is set to $H=10$ and the computational domain is chosen to be the cube of $[0, 10] \times [0, 10] \times [0, 10]$. The speed of the moving upper wall ($y=10$) in the x direction is $U=0.3$. The temperatures of the lower wall ($y=0$) and the upper one are $T_0=0.8$ and $T_1=0.85$ respectively. The analytical solution for this case is

$$\begin{aligned} u &= \frac{y}{H}U, \quad v = 0, \quad w = 0, \\ T &= T_0 + \frac{y}{H}(T_1 - T_0) + \frac{\mu U^2}{2k} \frac{y}{H} \left(1 - \frac{y}{H}\right), \\ p &= p_0, \quad \rho = \frac{\mathcal{P}}{T}, \end{aligned} \tag{5.2}$$

where γ is specific heat ratio and k is thermal conductivity. The static pressure is set to $p_0=1/\gamma$ and the viscosity of the fluid is assumed to be $\mu=0.01$. The flow variables at boundary faces are simply fixed to the exact solution.

Three successively refined prism grids are generated with $N=2, 4$ and 8 in the same way as in the vortex propagation case. Each cube is split into two prisms by the plane which is perpendicular to the $y=0$ plane. The error norms for the BR2 formulation are presented in Table 3. The density is used to evaluate the error. It is shown that nearly the formal order of accuracy is achieved for the 2nd- to 4th-order schemes.

5.2.2. Boundary Layer Flow over a Flat Plate

The laminar boundary layer flow over a plate is then computed using the CPR method. The Reynolds number based on the plate length is $Re_x = 10,000$ and the freestream Mach number is $M=0.2$. The plate length L is set to 1. The boundary layer thickness at the trailing edge is estimated by the approximate relation $\delta = 5L/\sqrt{Re_x}$. The computational domain is selected to be $(-2 \leq x \leq 1, 0 \leq y \leq 100\delta, -\delta \leq z \leq \delta)$. Note that the domain size in the y -direction is chosen to be large enough not to affect the results especially in the v -velocity profiles. The freestream values are specified at the inflow boundary at $x=-2$ and the top boundary at $y=100\delta$. For the lower boundary at $y=0$, the symmetry conditions are used on the upwind side to the wall ($-2 \leq x \leq 0$) and the adiabatic wall conditions are imposed on the wall ($0 \leq x \leq 1$). At the outflow boundary at $x=1$, only static pressure is prescribed. On the side boundaries at $z=0$ and δ , the symmetric conditions are assumed. First, we generated a three dimensional Cartesian mesh. The grid cells are clustered near the leading edge and the cell sizes are increased geometrically in both x - and y -directions. In the spanwise z -direction, we generate only one cell. Then we divide each hexahedral cell into two prisms to obtain a purely prismatic grid.

The computed u and v velocity profiles are compared with the Blasius's solution in Fig. 6. The computational grid used for the computations is generated to have 4 cells in the boundary layer at $x=1.0$ and 13 cells along the plate. The solution is apparently getting more accurate with the increasing of the order of polynomial approximation, and it is more clearly shown in the comparison of v -profiles. The computed skin friction coefficients on the wall are also plotted at Figure 7. The agreement with the Blasius's solution also becomes better with p -order refinement.

One of the concerning issues when we apply CFD solver to engineering problems is the stiffness arising from using high aspect ratio cells that are clustered near the solid wall to resolve the boundary layer especially in high Reynolds number flows. Reynolds numbers appearing in aerospace flow problems usually become $\sim 10^6$ or more, and so even if we make use of an implicit time integration scheme for numerical simulations, we will likely encounter still small time step restriction or deteriorated convergence rate. A possible remedy for this problem is employing a line solver [27,8]. Here we consider another approach to alleviate the stiffness issue by employing higher-order prism elements rather than having large number of lower order elements in the boundary layer. Since we use a tensor basis polynomial in prisms, we can use higher order polynomial only in the normal direction to the wall while using lower order one in the tangential directions to the wall so as to prevent the unnecessary increase of the computational cost.

Figure 8 shows the computed Mach number by using polynomials of degree 5 in the y -direction and polynomials of degree 2 in x - and z - directions. The grid has only two cells in the boundary layer at $x=1.0$ and 17 cells along the plate. The numbers of prism cells and DOFs are 728 and 26208 respectively. For comparison, we generated another grid that has more cells in the boundary (8 cells at $x=1.0$) but the same resolution in the x - and z -directions and employed degree 2 polynomials in all directions, resulting 1736 prisms and 31248 DOFs. In Fig. 9, the computed v -velocity profiles are shown. The computed profiles agree well with each other and also with the Blasius's solution. The convergence histories are compared in Fig. 10. The computations were performed using the LU-SGS scheme with the same time step. Compared to the computation using the lower order scheme with the finer grid, employing the higher order scheme with less grid cells gave the reductions of about 38% and 30 % in terms of Time steps and CPU times to reach machine zero residual, although the DOFs are about 16% less than the other's one.

5.2.3. Steady Subsonic Flow over a Sphere at $Re=118$

A steady viscous flow around a sphere is computed to validate the developed NS solver on a full 3D mixed mesh. The Reynolds number based on the diameter was chosen to be 118 so that we can compare the obtained results with experimental data [37] and numerical results using the SD scheme [36,44]. The Mach number is 0.2535 that is the same value in the reference computations. The mesh is generated to have five layers of prism cells and isotropic tetrahedral cells for the remaining region. We plot the cut of the grid on a plane with $y=0$ and surface mesh on the sphere in Fig. 11. The total number of mixed cells is 24,334.

The computations were performed using the 2nd- to 4th-order schemes. The computed Mach number contours and streamlines near wake using the 4th-order CPR scheme are shown in Fig. 12 and Fig. 13, respectively. We confirmed that the computed streamlines and the size of separation region agree well with both the experimental picture and the numerical results in the references. Here we only show a comparison of the computed skin friction profiles at the cross section ($y=0$) of the sphere in Fig. 14. The skin friction coefficients computed by the 4th-order CPR scheme and the 6th-order SD scheme are right on top of each other. The 3rd-order CPR result also agrees well with other results, though one can see only minor differences between those profiles. The predicted separation angle using the 4th-order CPR scheme is 123.6 deg (the wind side stagnation point has an angle of 0), which is identical to the value predicted by the 6th-order SD scheme. In Fig. 15, the computed drag coefficient by 4th-order CPR is compared to available experimental data. The agreement is also very good.

5.2.4. Unsteady Subsonic Flow over a Sphere at $Re=300$

We consider an unsteady flow case over the sphere with radius $r=1$ at the Reynolds number of 300 based on the diameter of the sphere. The inflow Mach number is assumed to be 0.3 in this case. The computational mesh is shown Fig 16. To resolve shedding vortices, the mesh is generated to have finer cells in the wake region. The total number of mixed cells is 54,312. Local grid size around the sphere is ~ 0.2 and the size in the wake region is ~ 0.8 . In this case, we employed the 3rd-order TVD Runge-Kutta method for the time integration and computed by the MPI parallelized code using 8 cores of a cluster machine to reduce the wall clock time.

The computed Q isosurface colored by local Mach number using the 4th-order CPR scheme is shown in Fig. 17.

The obtained plain symmetric wake vortex structure is comparable to the available experimental and computational results in [10,20] at least qualitatively. In Fig. 18 we plot the history of the drag coefficient C_d in terms of non-dimensional time t . The computed drag coefficient and the oscillating amplitude of drag and the Strouhal number Str are shown in the Table 4. For comparison, results from Gassner [10] using the 4th-order DG scheme on tetrahedral grid and from Tomboulides [38] and Johnson and Patel [20] obtained by incompressible simulation, are shown as well. The results computed by the CPR method reasonably agree with those reference values.

Table 1. Test of CPR-DG for vortex propagation problem (tetrahedral grids).

<i>Polynomial degree k</i>	<i>Grid size</i>	L_1 error	L_1 order	L_∞ error	L_∞ order
1	10x10x10x6	5.23e-3	-	9.56e-2	-
	20x20x20x6	1.42e-3	1.88	3.57e-2	1.42
	40x40x40x6	3.43e-4	2.05	9.76e-3	1.87
2	10x10x10x6	1.68e-3	-	6.06e-2	-
	20x20x20x6	2.61e-4	2.69	1.19e-2	2.35
	40x40x40x6	3.77e-5	2.79	1.51e-3	2.98
3	10x10x10x6	4.00e-4	-	2.05e-2	-
	20x20x20x6	2.44e-5	4.04	1.67e-3	3.62
	40x40x40x6	1.33e-6	4.20	1.00e-4	4.06
5	10x10x10x6	5.66e-5		2.34e-3	
	20x20x20x6	9.70e-7	5.87	7.78e-5	4.91

Table 2. Test of CPR-DG for vortex propagation problem (prismatic grids).

<i>Polynomial degree k</i>	<i>Grid</i>	L_1 error	L_1 order	L_∞ error	L_∞ order
1	10x10x10x2	7.37e-3	-	1.34e-1	-
	20x20x20x2	2.12e-3	1.80	4.85e-2	1.47
	40x40x40x2	5.19e-4	2.03	1.19e-2	2.03
2	10x10x10x2	2.17e-3	-	4.77e-2	-
	20x20x20x2	2.67e-4	3.02	8.65e-3	2.46
	40x40x40x2	2.88e-5	3.21	1.04e-3	3.06
3	10x10x10x2	4.36e-4	-	1.54e-2	-
	20x20x20x2	2.70e-5	4.01	1.43e-3	3.43
	40x40x40x2	1.64e-6	4.04	9.38e-5	3.93

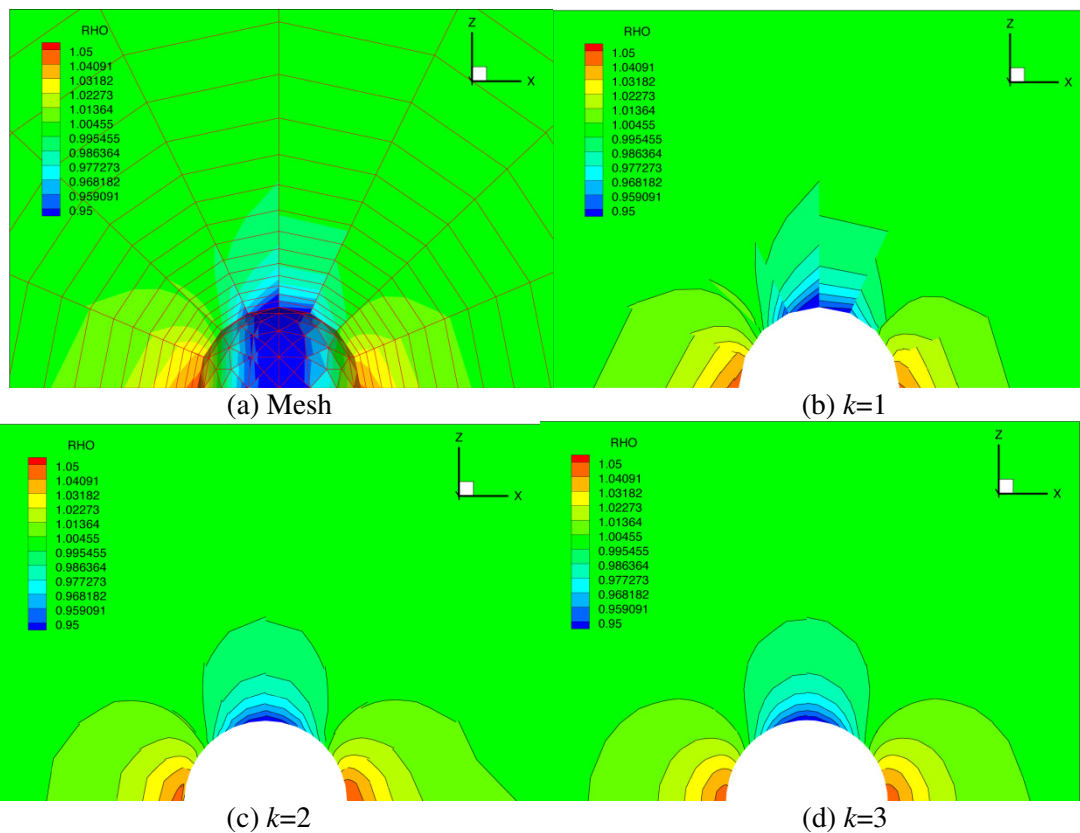


Figure 4. Prismatic grid and computed density contours for flow around a sphere.

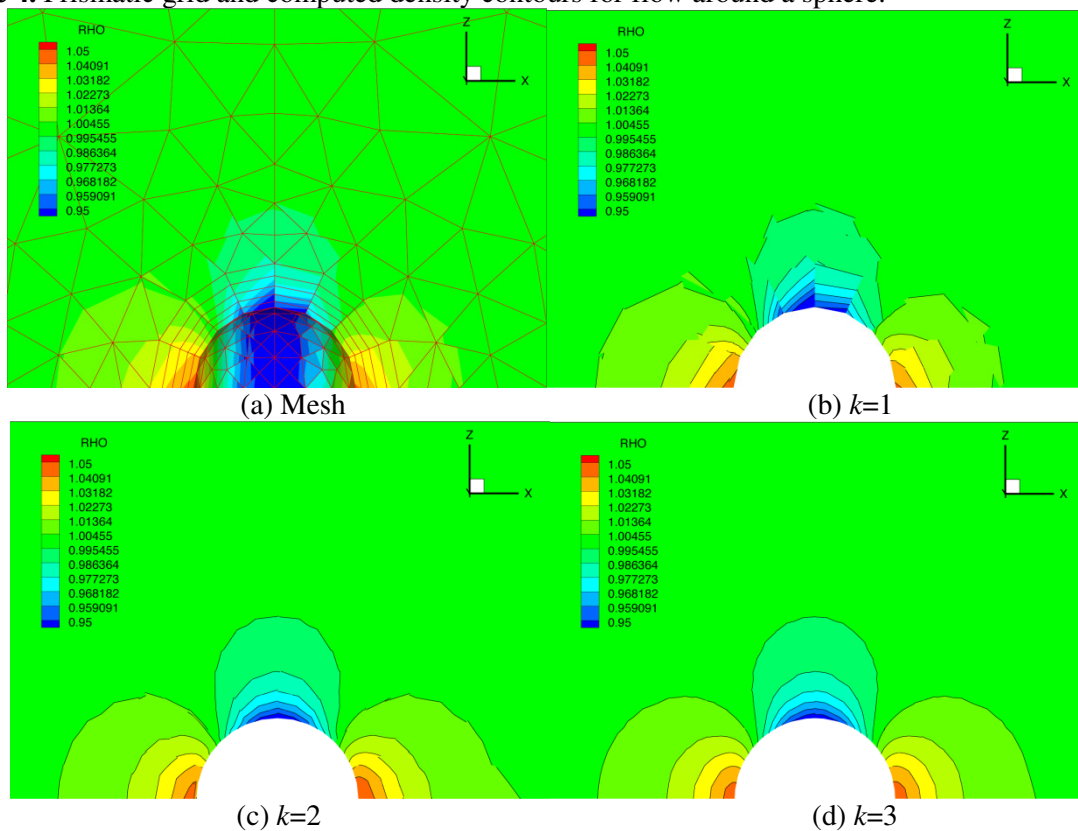


Figure 5. Mixed grid (tetrahedrons and prisms) and computed density contours for flow around a sphere.

Table 3. Test of CPR-DG (BR2) for Couette flow problem (prismatic grids).

<i>Polynomial degree k</i>	<i>Grid</i>	L_1 error	L_1 order	L_∞ error	L_∞ order
1	2x2x2x2	5.5525e-4	-	2.4030e-3	-
	4x4x4x2	1.1909e-4	2.221	3.9968e-4	2.588
	8x8x8x2	3.1063e-5	1.939	1.1574e-4	1.788
2	2x2x2x2	8.1732e-6	-	2.0928e-5	-
	4x4x4x2	1.2867e-6	2.667	3.3742e-6	2.633
	8x8x8x2	1.6758e-7	2.941	5.4916e-7	2.619
3	2x2x2x2	2.6248e-7	-	8.1984e-7	-
	4x4x4x2	2.0331e-8	3.690	5.7014e-8	3.846
	8x8x8x2	1.3907e-9	3.870	4.2087e-9	3.760

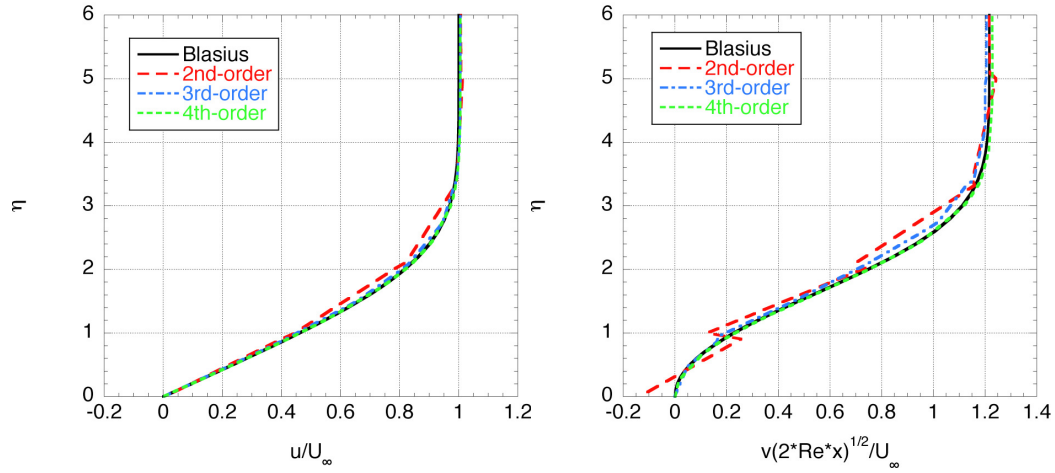


Figure 6. Comparisons of velocity profiles in the boundary layer at $x=0.5$. u - and v -profiles on the left and right.

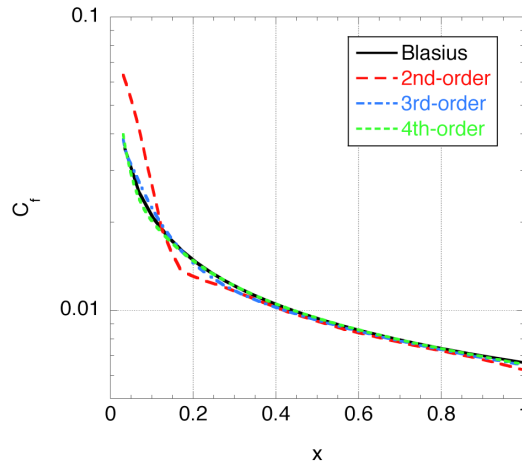


Figure 7. Comparison of the skin friction coefficient along the plate.

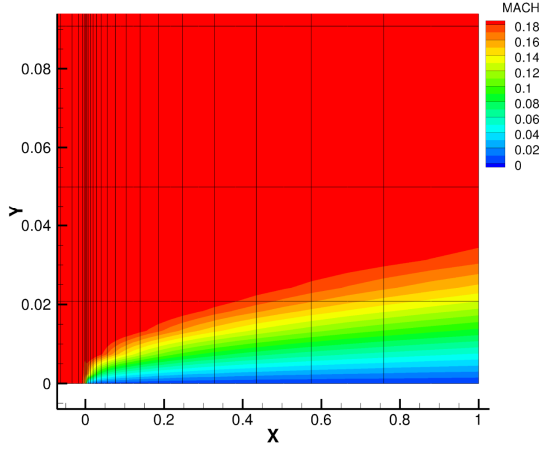


Figure 8. Grid and Mach number contours using the CPR scheme with polynomials of degree 5 in the y -direction (y direction stretched by factor 10).

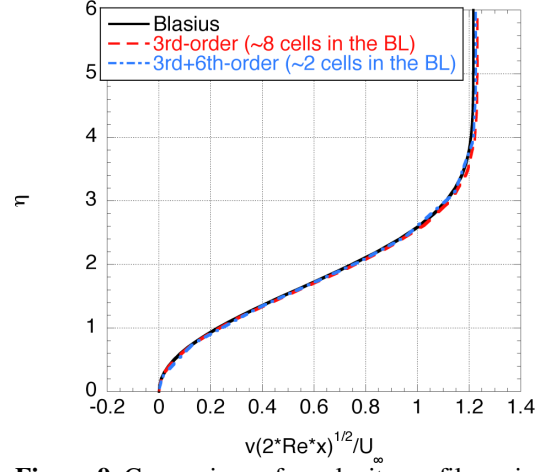


Figure 9. Comparison of v -velocity profiles using different degrees of polynomial and grids.

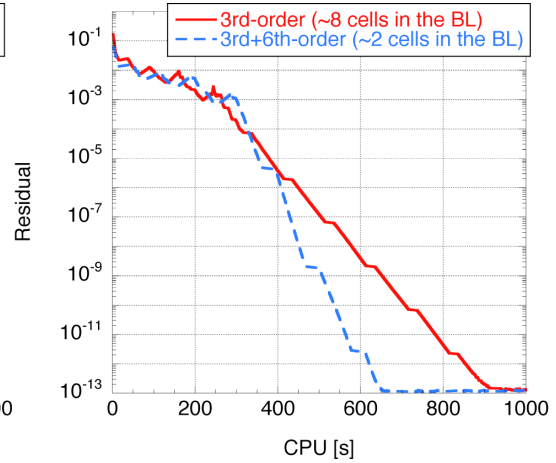
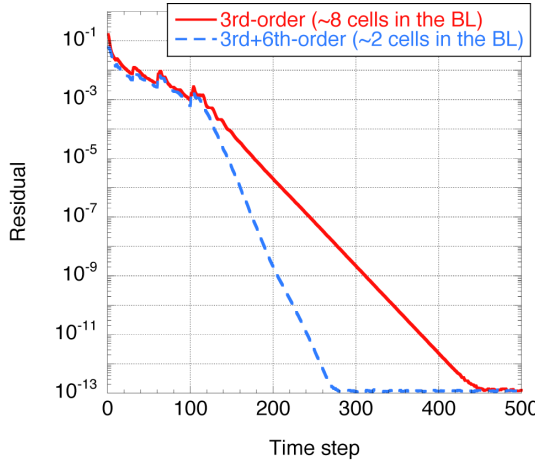


Figure 10. Comparisons of the convergence histories using different degrees of polynomial and grids.

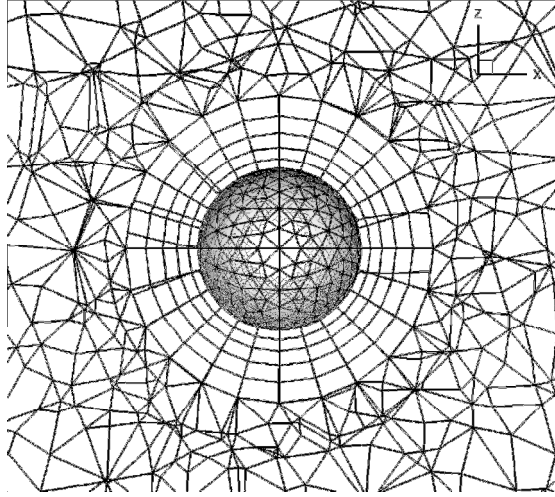


Figure 11. Computational grid around a sphere for the steady viscous flow over a sphere.

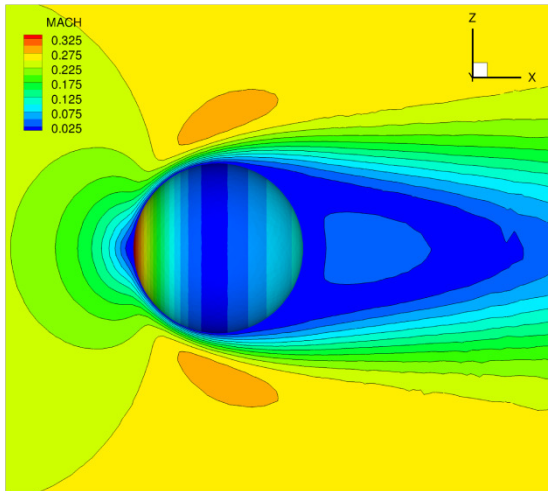


Figure 12. Computed pressure (on the sphere) and Mach number (on $y=0$ plane) distributions using the 4th-order CPR

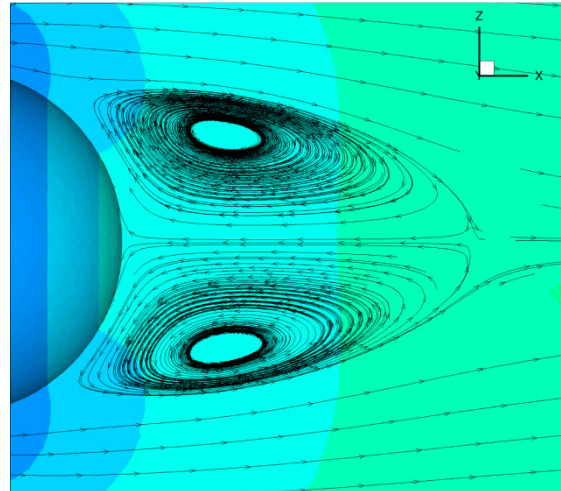


Figure 13. Computed streamlines using the 4th-order CPR scheme near the wake region behind the sphere.

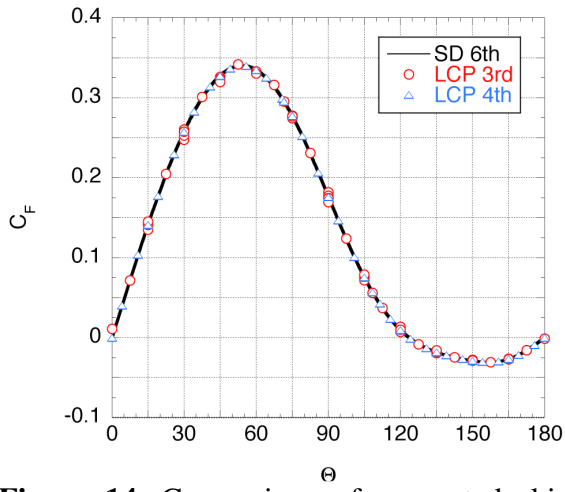


Figure 14. Comparison of computed skin friction coefficients using the 3rd- and 4th-order CPR schemes and 6th-order SD scheme[38]

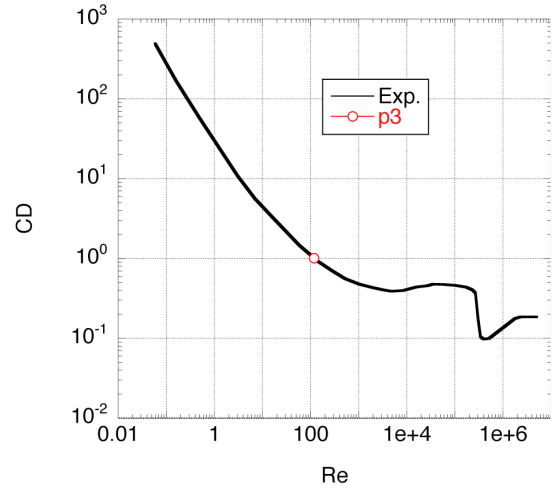


Figure 15. Comparison between the computed drag coefficient using the 4th-order CPR scheme and experimental data for a sphere

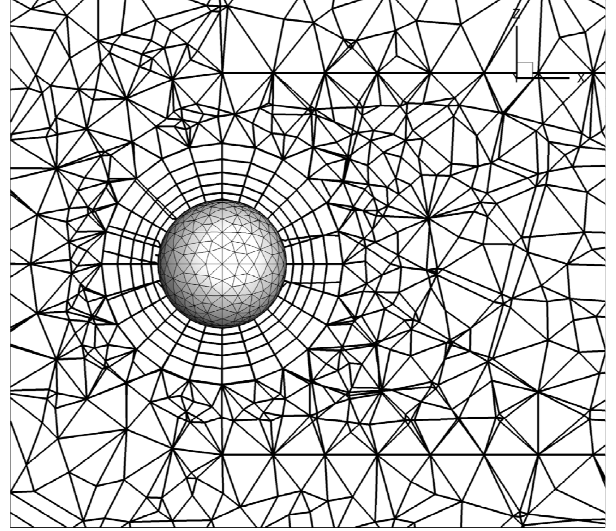
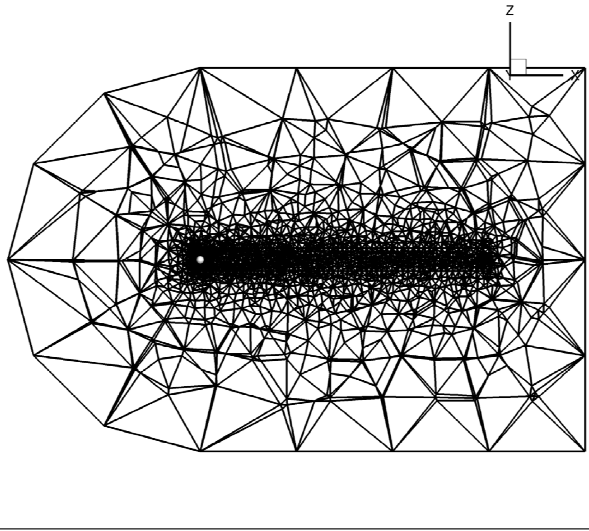


Figure 16. Computational grid around a sphere for the unsteady viscous flow over a sphere. (Left: Entire grid, Right: Grid around the sphere.)

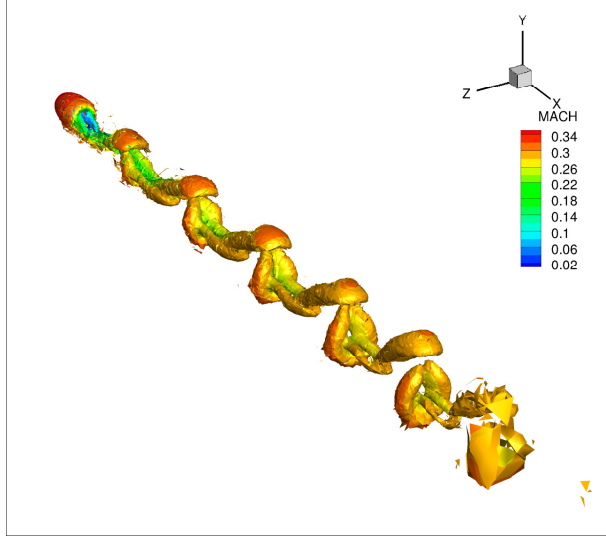


Figure 17. Computed Q isosurface in the wake region of the viscous laminar flow over a sphere at $Re=300$.

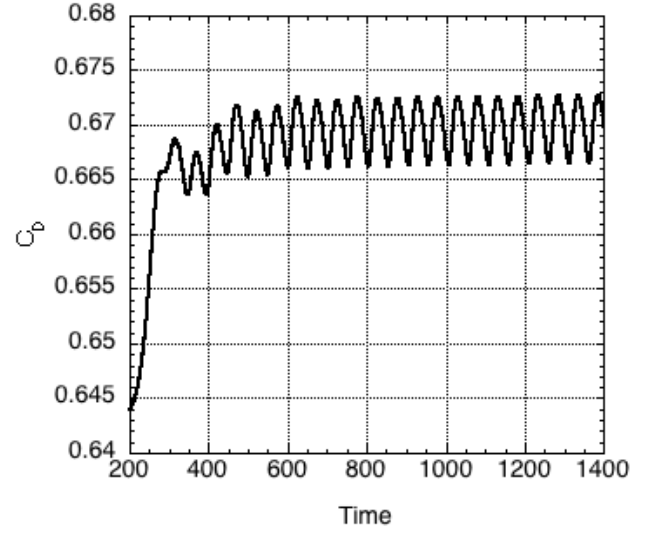


Figure 18. Time history of the drag coefficient.

Table 4. Comparisons of the averaged drag coefficient, the amplitude of drag and the Strouhal number.

Method	C_d	ΔC_d	Str
Present (LCP)	0.670	0.0032	0.131
Gassner [10]	0.673	0.0031	0.135
Tomboulides [38]	0.671	0.0028	0.136
Johnson & Patel [20]	0.656	0.0035	0.137

6. Conclusion

The CPR method is successfully extended to 3D hybrid unstructured meshes using tetrahedral and prismatic elements. The CPR formulation for tetrahedral elements is directly derived in the same manner for 2D triangular elements and the one for prism is obtained by just a combination of the 1D and 2D schemes. The resulting scheme needs no explicit integrations and no data reconstructions. This numerical efficiency is more significant in 3D simulations in comparison to 2D simulations because numerical complexities involved in high-order quadratures and reconstructions rapidly increase in 3D.

The developed CPR scheme is verified with grid convergence studies for an inviscid flow and a viscous flow, indicating that the developed scheme is capable of achieving nearly the optimal order of accuracy. Then, several validation cases are computed for solving the 3D Euler equations and the 3D NS equations. The CPR method performs very well to obtain high-order accurate solution for all cases. Future studies include extension to adopt hexahedral and pyramidal cells for more flexible geometry discretizations and hp-adaptation techniques for realizing practical high accurate CFD simulations.

References

- [1] F. Bassi, S. Rebay, A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier–Stokes equations, *J. Comput. Phys.* 131 (1) (1997) 267–279.
- [2] F. Bassi, S. Rebay, GMRES discontinuous Galerkin solution of the compressible Navier–Stokes equations, in *Lecture Note in Computational Science and Engineering* 11, Springer Verlag, New York, (2000), 197–208.
- [3] Q. Chen, I. Babuska, Approximate optimal points for polynomial interpolation of real functions in an interval and in a triangle, *Comput. Methods Appl. Mech. Eng.* 128 (1995) 405–417.
- [4] B. Cockburn, S.-Y. Lin, C.-W. Shu, TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one-dimensional systems, *J. Comput. Phys.* 84 (1989) 90–113.
- [5] B. Cockburn, C.-W. Shu, TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws II: general framework, *Math. Comput.* 52 (1989) 411–435.
- [6] B. Cockburn, C.-W. Shu, The local discontinuous Galerkin methods for time-dependent convection diffusion systems, *SIAM J. Numer. Anal.* 35 (1998) 2440–2463.
- [7] B. Cockburn, C.-W. Shu, The Runge–Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems, *J. Comput. Phys.* 141 (1998) 199–224.
- [8] K.J. Fidkowski, T.A. Oliver, J. Lu, D.L. Darmofal, p-Multigrid solution of high-order discontinuous Galerkin discretizations of the compressible Navier-Stokes equations. *J Comput. Phys.* 207 (2005), 92–113.
- [9] H. Gao, Z.J. Wang, A high-order lifting collocation penalty formulation for the Navier-Stokes equations on 2D mixed grids, *AIAA Paper* 2009-3784.
- [10] G. J. Gassner, F. Lorcher, C-D. Munz, and J. S. Hesthaven, Polymorphic nodal elements and their application in discontinuous Galerkin methods, *J. Comput. Phys.* 228 (2009), 1573–1590.
- [11] S.K. Godunov, A finite-difference method for the numerical computation of discontinuous solutions of the equations of fluid dynamics, *Mat. Sb.* 47 (1959) 271.
- [12] T. Haga, M. Furudate, K. Sawada, RANS simulation using high-order spectral volume method on unstructured tetrahedral grids, *AIAA Paper*, 2009-404.
- [13] T. Haga, K. Sawada, Z.J. Wang, An implicit LU-SGS scheme for the spectral volume method on unstructured tetrahedral grids, *Communications in Computational Physics*, Vol. 6, No. 5, pp. 978–996 (2009).
- [14] R. Harris, Z.J. Wang, Y. Liu, Efficient quadrature-free high-order spectral volume method on unstructured grids: theory and 2D implementation, *J. Comput. Phys.* 227 (3) (2008) 1620–1642.
- [15] J.S. Hesthaven, From electrostatics to almost optimal nodal sets for polynomial interpolation in a simplex, *SIAM J. Numer. Anal.* 35 (2) (1998) 655–676.
- [16] J.S. Hesthaven, Tim Warburton, *Nodal Discontinuous Galerkin Methods*, Springer, 2008.
- [17] H.T. Huynh, A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods, *AIAA Paper* 2007-4079.
- [18] H.T. Huynh, A Reconstruction Approach to High-Order Schemes Including Discontinuous Galerkin for Diffusion, *AIAA Paper* 2009-403.

- [19] A. Jameson, Analysis and design of numerical schemes for gas dynamics. I. Artificial diffusion, upwind biasing, limiters and their effect on accuracy and multigrid convergence, *Int. J. Comput. Fluid Dyn.* 4 (1994) 171–218.
- [20] T. A. Johnson and V. C. Patel, Flow past a sphere up to a Reynolds number of 300, *J. Fluid Mech.* 378 (1999), 19-70.
- [21] D.A. Kopriva, J.H. Kolas, A conservative staggered-grid Chebyshev multidomain method for compressible flows, *J. Comput. Phys.* 125 (1996) 244.
- [22] M.-S. Liou, A sequel to AUSM, Part II: AUSM+-up for all speeds, *J. Comput. Phys.* 214 (2006) 137–170.
- [23] Y. Liu, M. Vinokur, Z.J. Wang, Discontinuous spectral difference method for conservation laws on unstructured grids, in: *Proceedings of the Third International Conference on Computational Fluid Dynamics*, Toronto, Canada, July 12–16, 2004.
- [24] Y. Liu, M. Vinokur, Z.J. Wang, Discontinuous spectral difference method for conservation laws on unstructured grids, *J. Comput. Phys.* 216 (2006) 780– 801.
- [25] Y. Liu, M. Vinokur, Z.J. Wang, Spectral (finite) volume method for conservation laws on unstructured grids V: extension to three-dimensional systems, *J. Comput. Phys.* 212 (2006) 454–472.
- [26] H. Luo, J. Baum and R. Lohner, A discontinuous Galerkin method based on a Taylor basis for the compressible flows on arbitrary grids, *J. Comput. Phys.* 227 (2008), 8875-8893.
- [27] D.J. Mavriplis, Multigrid strategies for viscous flow solvers on anisotropic unstructured meshes. *J. Comput. Phys.* 145 (1998) 141-165
- [28] G. May, A. Jameson, A spectral difference method for the Euler and Navier–Stokes equations, AIAA Paper No. 2006-304, 2006.
- [29] C.R. Nastase, D.J. Mavriplis, High-order discontinuous Galerkin methods using an hp-multigrid approach, *J. Comput. Phys.* 213 (2006) 330–357.
- [30] S. Osher, Riemann solvers, the entropy condition, and difference approximations, *SIAM J. Numer. Anal.* 21 (1984) 217–235.
- [31] W.H. Reed, T.R. Hill, Triangular mesh methods for the neutron transport equation, Los Alamos Scientific Laboratory Report, LA-UR-73-479, 1973.
- [32] P.L. Roe, Approximate Riemann solvers, parameter vectors, and difference schemes, *J. Comput. Phys.* 43 (1981) 357–372.
- [33] V.V. Rusanov, Calculation of interaction of non-steady shock waves with obstacles, *J. Comput. Math. Phys. USSR* 1 (1961) 261–279.
- [34] S. J. Sherwin, and G. E. Karniadakis, A new triangular and tetrahedral basis for high-order (hp) finite element methods, *Int. J. Num. Meth. Eng.*, 38 (1995), 3775-3802.
- [35] C.-W. Shu, Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws, in: B. Cockburn, C. Johnson, C.-W. Shu, E. Tadmor (Eds.), *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*, in: A. Quarteroni (Ed.), *Lecture Notes in Mathematics*, vol. 1697, Springer, 1998, pp. 325–432.
- [36] Y. Sun, Z.J. Wang and Y. Liu, High-Order Multidomain Spectral Difference Method for the Navier-Stokes Equations on Unstructured Hexahedral Grids, *Communications in Computational Physics*, Vol. 2, No. 2, pp. 310-333 (2007).
- [37] S. Taneda, Experimental investigations of the wake behind a sphere at low Reynolds numbers, *J. Phys. Soc. Japan*, 11 (1956), 1104-1108.
- [38] A. G. Tomboulides and S. A. Orszag, Numerical investigation of transitional and weak turbulent flow past a sphere, *J. Fluid Mech.* 416 (2000), 45-73.

- [39] K. Van den Abeele, C. Lacor, An accuracy and stability study of the 2D spectral volume method, *J. Comput. Phys.* 226 (1) (2007) 1007–1026.
- [40] K. Van den Abeele, C. Lacor, Z.J. Wang, On the stability and accuracy of the spectral difference method, *J. Sci. Comput.* 37 (2) (2008) 162–188.
- [41] B. van Leer, Towards the ultimate conservative difference scheme V. A second order sequel to Godunov’s method, *J. Comput. Phys.* 32 (1979) 101–136.
- [42] B. Van Leer, S. Nomura, Discontinuous Galerkin for diffusion, AIAA Paper No. 2005-5108, 2005.
- [43] Z.J. Wang, Spectral (finite) volume method for conservation laws on unstructured grids: basic formulation, *J. Comput. Phys.* 178 (2002) 210–251.
- [44] Z.J. Wang, High-order methods for the Euler and Navier–Stokes equations on unstructured grids, *J. Prog. Aerosp. Sci.* 43 (2007) 1–47.
- [45] Z.J. Wang and H. Gao, A unifying lifting collocation penalty formulation including the discontinuous Galerkin, spectral volume/difference methods for conservation laws on mixed grids, *J. Comput. Phys.* 228 (2009) 8161–8186.
- [46] Z.J. Wang, Y. Liu, Spectral (finite) volume method for conservation laws on unstructured grids II: extension to two-dimensional scalar equation, *J. Comput. Phys.* 179 (2002) 665–697.
- [47] Z.J. Wang, Y. Liu, Spectral (finite) volume method for conservation laws on unstructured grids III: one-dimensional systems and partition optimization, *J. Sci. Comput.* 20 (1) (2004) 137–157.
- [48] Z.J. Wang, L. Zhang, Y. Liu, Spectral (finite) volume method for conservation laws on unstructured grids IV: extension to two-dimensional Euler equations, *J. Comput. Phys.* 194 (2) (2004) 716–741.
- [49] T. Warburton, An explicit construction of interpolation nodes on the simplex, *J. Eng. Math.* 56 (2006) 247–262.
- [50] O.C. Zienkiewicz, R.L. Taylor, *The Finite Element Method The Basics*, vol. 1, Butterworth–Heinemann, Oxford, England, 2000.